VA-119-001-06-TR

SERDP SEED PROJECT UX-1449 FINAL REPORT

MONTE CARLO ENGINE FOR EMI SURVEY ANALYSIS

September 2006

PERFORMING ORGANIZATION

AETC Incorporated
1225 South Clark Street, Suite 800
Arlington, VA 22202

PRINCIPAL INVESTIGATOR
Jonathan Miller

This research was supported wholly by the U.S. Department of Defense, through the Strategic Environmental Research and Development Program (SERDP) through Project UX-1449 under Contract W912HQ-05-P-0038.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 SEP 2006	**Final**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Monte Carlo Engine for EMI Survey Analysis	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Jonathan Miller**	5d. PROJECT NUMBER **MM-1449**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **AETC Incorporated 1225 South Clark Street, Suite 800 Arlington, VA 22202**	8. PERFORMING ORGANIZATION REPORT NUMBER **VA-119-001-06-TR**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **Strategic Environmental Research & Development Program 901 North Stuart Street, Suite 303 Arlington, VA 22203**	10. SPONSOR/MONITOR'S ACRONYM(S) **SERDP**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**36**	

Table of Contents

List of Figures

List of Tables

Acknowledgments

This project was funded fully by SERDP under project UX-1449 through contract W912HQ-05-P-0038.

Executive Summary

This project produced and demonstrated proof-of-principle for a Monte Carlo tool that can calculate performance measures under any given set of survey conditions and analysis methods. The existing Monte Carlo tool at AETC was improved by incorporating more realistic noise models, inherent variability of UXO items, and the ability to utilize different discrimination algorithms. The tool was used to show the potential improvement of a hybrid approach to discrimination analysis over an unconstrained or weighted unconstrained approach. It was also used to investigate sensitivities to field survey conditions and noises and, even in the limited proof-of-principle runs, clear guidance on the strong effect of at least one system parameter (timing error) was obtained.

Signal-to-noise ratio is a critical parameter for successful UXO discrimination, and accurate noise models are a key part of any Monte Carlo analysis. In this project we improved existing noise models by incorporating correlation scales observed from field data. These field data, however, typically contain only aggregate information, which makes it difficult to discover the magnitude of the various components involved. ESTCP project MM-0508 "Quantification of Noise Sources in EMI Surveys" is aimed at producing the data which will make these component determinations possible, and in fact we have used some preliminary data from that project in this work.

At the start of each iteration in the Monte Carlo code, target response values (beta values) were randomly drawn from a library and synthetic data was produced using the dipole model. These synthetic data did not, therefore, exhibit non-dipolar effects, something which could be incorporated in future work using a more sophisticated forward model.

Beta values were drawn from a list of 98 possible targets, representing four UXO types: 20mm, 60mm mortar, 81mm mortar, and 3 inch Stokes mortar. These betas were taken directly from test stand data on real targets, and reflect the variability which occurs within UXO classes. Future work could expand this list to include a wider assortment of UXO and clutter items.

The focus of this project was on evaluation of operational survey parameters such as lane spacing and vehicle speed, in terms of their impact on the ability to recover accurate betas. We do not expect the omission of clutter to affect these evaluations, since performance was judged on comparison of recovered values against library values used to represent the class. This approach allows us to introduce inherent UXO variability by utilizing “true” betas in data synthesis which differ slightly from the “class archetype” betas used in data analysis. This is a realistic environment in which to test inversion success.

1. Project Objective

There is need for detailed guidance on implementation of EMI surveys, including specific operational parameters such as optimal data densities, vehicle speeds, and the cost-benefit of cued re-surveying of targets [1]. The overall aim of this project was to produce a tool that, given a set of survey conditions and methods, calculates performance measures. This tool allows survey methods to be tested with a computer and the results will produce suggestions for improved operations on real sites. The goal of this project is to show proof-of-principle that the Monte Carlo tool, consisting of a carefully constructed simulator combined with accurate models, can provide the relevant information, including trade-offs involved, for all major components of the survey process. In the future, it can also be useful for developing QA/QC protocols, evaluating performance on two or more overlapping targets, and quantifying the relative effectiveness of new models for data inversion.

2. Background

Modern electromagnetic geophysical surveys are typically conducted under GPS control using arrays of magnetometers or EMI sensors. Typical vehicular towed array surveys produce very high-density data maps of 200,000-2,000,000 data points per acre when using EMI and magnetometer sensors. Target analyses typically fit perceived signal anomalies to dipole signature models. To improve the ability to distinguish intact UXO from metallic scrap, statistical analysis approaches are often applied to the output parameters of the physics-based target-fitting algorithms. Although we can approach 100% detection of the UXO threats on fairly uncomplicated ranges, the current classification capability and resulting false alarm rates mean that clearing the ranges requires digging 5-25 non-UXO targets to recover each intact UXO.

Successful detection and classification of buried targets depends strongly on the quality of data available from the geophysical survey. Data density, motion noise, and navigation errors are a few of the many factors that govern data quality, but the relationship between these factors and operational survey practices (e.g., lane spacing, sensor configuration, and vehicle speed) are complex and not well understood. In this research project, we aim to apply the natural advantages of Monte Carlo simulation to allow precise investigation of these relationships without loss of flexibility in defining the behavior of each system component.

The name *Monte-Carlo method* is given to all procedures that make use of randomness for the solution of deterministic problems. This approach revolutionized many fields of modern experimental science because it allows arbitrarily precise estimation of unknown system parameters whose direct calculation is not tractable. Results are given in the form of histograms generated from repeated experiments in which random realizations are processed.

3. Technical Methods

3.1 Monte Carlo Integration

The problem of evaluating performance of EMI surveys may be cast as a multi-dimensional integration. We are interested in evaluating some performance measure, say probability of detection *Pdet*, under a variety of conditions, including target depth, target orientation, noise, data locations, navigation errors, etc. These factors can be thought of as a very long list of variables, $\boldsymbol{X} = \{x_1, x_2, ..., _{n-1}, x_n\}$, representing such things as the sensor tilt at a particular location, or the y coordinate error of another, etc. The variables $\boldsymbol{X}$ are generally not independent, and their joint probability density function $pdf(\boldsymbol{X})$ must be known or assumed in order to evaluate *Pdet*. For any given case $\boldsymbol{X}_0$, the target will either be detected or not, so the deterministic function $I = f(\boldsymbol{X})$ may be defined such that $I = 1$ when the target is detected, and $I = 0$ otherwise. *Pdet* is then:

$$Pdet = \iiint ... \iiint I(\boldsymbol{X}) pdf(\boldsymbol{X}) d\boldsymbol{X} . \qquad (1)$$

This integrand is bounded in the interval (0,1), has multiple discontinuities, and the boundaries on the running variables are complicated. This makes the problem a good candidate for *Monte Carlo Integration* [2]. Monte Carlo Integration evaluates the integrand at a random sample of points, and estimates its integral based on that random sample.

The widely acclaimed text *Numerical Recipes*, states "Offered the choice between mastery of a five-foot shelf of analytical statistics books and middling ability at performing statistical Monte Carlo simulations, we would surely choose to have the latter skill!" [2] p.691. Monte Carlo methods are extremely robust and flexible, especially with the increased computational power available today, and they have revolutionized many fields of modern experimental science since their introduction in the 1950s. They are particularly useful for studying systems with a large number of degrees of freedom, such as the EMI field surveys being considered here.

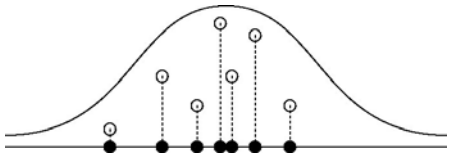

Figure 1. Uniform spatial sampling under the curve (open dots) is equivalent to weighted sampling using the *pdf* (solid dots).

We implement the Monte Carlo tool by creating *N* realizations drawn randomly from the joint probability density function $pdf(\boldsymbol{X})$, with each realization representing a realistic example of a buried object along with sensor survey lanes crossing it and associated signals. These synthetic data are submitted to inversion using the same procedure as on real data, and the indicator

function $I(\boldsymbol{X})$ is evaluated, representing success or failure (1 or 0) of the performance measure (e.g., detection or discrimination). We note that sampling the domain of $\boldsymbol{X}$ in this way, using $pdf(\boldsymbol{X})$, is identical to uniform sampling in the higher-dimensional space that incorporates $pdf(\boldsymbol{X})$ in additional dimensions (Figure 1). This allows precise application of the basic theorem of Monte Carlo integration:

$$Pdet \approx \langle I \rangle \pm \sqrt{\frac{\langle I^2 \rangle - \langle I \rangle^2}{N}}\ , \tag{2}$$

where the angle brackets denote the arithmetic mean over the N sample points:

$$\langle I \rangle \equiv \frac{1}{N}\sum_{i=1}^{N} I(\boldsymbol{X}_i) \qquad \langle I^2 \rangle \equiv \frac{1}{N}\sum_{i=1}^{N} I^2(\boldsymbol{X}_i)\ . \tag{3}$$

The plus or minus term in Equations 2 and 3 above is a one standard deviation error estimate for the integral, not a rigorous bound. Furthermore, there is no guarantee that the error is distributed as a Gaussian, so the error term is only a rough indication of probable error.

3.2 The Number of Runs *N*

It is clear from equation 2 that increasing N improves the accuracy of the estimate. However increasing N also lengthens computation time, which reduces the overall number of different configurations that can be investigated, so there is a trade-off.

During the process of developing the Monte Carlo code, preliminary runs were made with N=50, N=100, and N=200. Those runs do not bear direct comparison against the final results because parameter inputs had not been finalized, so noise levels and default survey configurations were different, but results suggested that N=100 would deliver estimates within about 10% of the true (N=infinity) answer in most cases. Since we were aiming to discover large trends across many variables, we felt 10% was acceptable, and settled on N=100.

The following example shows application of equations 2 and 3, and illustrates the effect of N. If we define a decision rule, for example by choosing a threshold on some output of inversion, e.g. drawing a horizontal line through values plotted in figure 11, then the success rate is estimated using equation 2, where the indicator function $I(X)$ evaluates to 1 for trials in which correct identification (red diamonds on figure 11) occurred within the decision rule (below the threshold) and 0 otherwise. If, for example, the threshold includes 75% of correct ID's then the estimated success rate, including confidence interval, is (from equation 2):

$$P_{success} = 0.75 \pm \sqrt{\frac{0.75 - 0.56}{100}} = 0.75 \pm 0.043.$$

So the success rate of this decision rule is estimated to be 75%, within ±4% at N=100. Note that errors attenuate like $1/\sqrt{N}$, so there are diminishing returns with increasing N. A ten-fold increase, to N=1000, only improves accuracy to ±1.4%. These considerations led us to choose N=100 as our default.

Note that the zero values of $I(X)$ increase the error estimate term, as expected since points outside the decision rule contain less information about the rule, as discussed in [2] p.305. For this reason, it is beneficial to evaluate the inverse of the decision rule for cases with low success rates. This maneuver is possible only because the domain from which we draw random realizations of individual survey anomalies is assumed to approximate the entire range of possibilities.

3.3 Monte Carlo Improvements

This Monte Carlo Code represents an improved version of an earlier code that was used in SERDP project CU-1121 [3]. There have been several improvements, including the incorporation of UXO variability in the EMI response, and adding realistic correlation structure to noise components. We also benefited from having more field data from parallel research efforts, which allow more realistic estimation of noise levels.

3.3.1 Incorporation of UXO variability

Four UXO classes were included in this study: 20mm projectiles, 60mm mortars, 81mm mortars, and 3inch Stokes mortars. Using the EMI response database collected under SERDP project UX-1313, which investigated the inherent variability of UXO targets, a group of betas derived from test stand measurements on real UXO were gathered for each class. Table 1 shows how many in each class were included in this effort. Each item contributed two possible beta sets because the nose-up and nose-down signals differ, due to the non-uniformity of the primary field. Figure 2 shows a plot of the longitudinal betas, for both nose-up and nose-down configuration, used for the different targets. The substantial spread indicates the magnitude of UXO variability as a contributor to overall noise in the problem. With each iteration of the Monte Carlo loop, we select one random UXO item from the database (i.e., set of UXO in the same class) and use the associated beta values to generate synthetic data to create a hypothetical UXO with realistic attributes.

UXO class	Count
20mm	21
60mm	30
81mm	18
3in Stokes	29

Table 1. Number of individual UXO in each class from which to choose betas.

3.3.2 Improved Noise Models

We defined correlated random processes to represent (1) roll and pitch of the sensor, (2) error in roll and pitch measurement, (3) vertical "bounce" and transverse "meander" deviations from intended straight survey lanes, (4) error in GPS x, y, & z position readings, (5) geologic noise, (6) timing error between GPS and sensor signals, and (7) sensor noise.

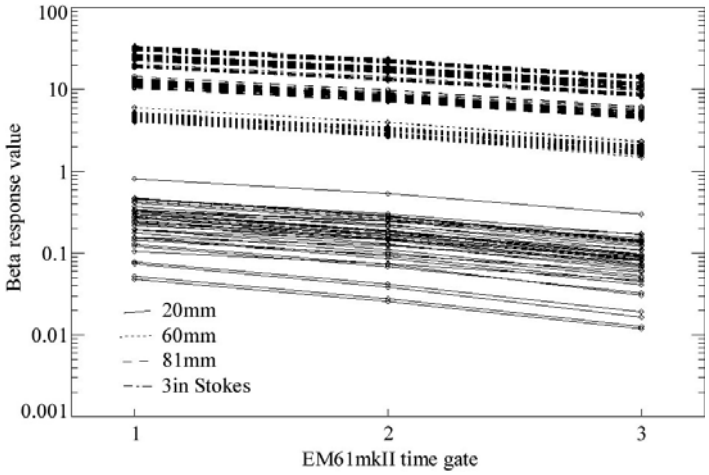

Figure 2. Longitudinal response betas for a group of UXO. These were selected at random in order to incorporate realistic UXO variability in the generation of synthetic data.

3.3.3 Noise correlation

Noise components in EMI field survey data arise from random processes with potentially complex correlation structures, including geologic noise and GPS navigation errors. To improve our Monte Carlo simulations, we have made an effort to match the correlation lengths of our synthetic noise components with those of real data. The correlation length, θ, of a random process, also called the scale of fluctuation, is defined [4] as the area under the correlation function $\rho(\tau)$, and it is also proportional to the unit-area one-sided spectral density function $g(\omega)$ at zero frequency:

$$\theta = 2\int_0^\infty \rho(\tau)d\tau \qquad\qquad \theta = \pi\, g(0)\,. \tag{4}$$

Correlation length is a compact and easily understood first-order descriptor of a random process and it is possible to estimate θ for natural processes by simple inspection of sample correlation graphs. Although θ cannot be defined for certain random processes (e.g., self-similar or fractal processes), good approximations can always be found in locally bounded domains such as EMI survey sites.

We use this approach instead of matching sample spectral densities directly because correlation scales provide an easy way to evaluate data requirements: if the apparent correlation scale is long compared to the length of the data record, then more data is needed. In any case, correlation $\rho(\tau)$ is directly related to spectral density $g(\omega)$ through the Wiener-Khinchine relations:

$$\rho(\tau) = \int_0^\infty g(\omega)\cos(\omega\tau)d\omega$$
$$g(\omega) = \frac{2}{\pi}\int_0^\infty \rho(\tau)\cos(\omega\tau)d\tau \quad , \qquad (5)$$

so that fitting a correlation model to observed correlation data is equivalent to fitting the associated spectral density model to the observed spectral density data. We assume triangular correlation for $\rho(\tau)$ in all cases for simplicity, and also because many of the records are not long enough to support more elaborate models.

The subject of modeling correlation structure in natural phenomenon has been heavily researched, and it is still regarded as an inexact science. In practice, complicated models of correlation do not lead to better estimates compared to simpler ones. If a simple model captures all the main features of observed sample correlation, then it will generally provide estimates that are just as accurate as those found from a complex one. This conclusion can be reached a number of ways, including cross validation of data. For discussion of this point, see [5] starting on p. 377. A comprehensive discussion of random field modeling is found in [4], and certain subtle aspects are addressed in [6].

Triangular correlation results when a moving average is performed on an underlying random process, provided the averaging window is roughly rectangular, and the length of the window is long (about twice as long or more) compared to the correlation length of the underlying process [4] p.202. So it is reasonable to expect that triangular correlation would appear, and perhaps dominate, in natural systems that involve averaging, such as GPS position data or platform motion.

Gaussian correlation is characterized by a change of curvature close to zero lag, so that correlation flattens out as it crosses the origin. None of the graphs of sample correlation (figures 3,4 and 5) suggest this kind of model. Absent a compelling physical justification for Gaussian correlation, and absent any indication of it in the data, we feel justified in not using it.

To be clear, our synthetic fields have Gaussian distribution at each point, i.e. the individual field values come from a normal distribution. But the join density functions are not multivariate normal: they are governed instead by triangular correlation, forming a complete definition of the random field model. Synthetic realizations are constructed as a moving average of *m* uncorrelated observations of a white noise process. The resulting correlation function *ρ(τ)* is triangular,

$$\rho(\tau) = \begin{cases} 1 - |\tau|/m, & |\tau| \le m \\ 0, & |\tau| \ge m \end{cases}, \qquad (6)$$

the scale of fluctuation is equal to *m*, and the point variance is

$$\sigma_m^2 = \sigma^2 / m \,. \tag{7}$$

We have included the correlation structure of the following noises in the Monte Carlo code: geologic noise, timing error between EMI signal and navigation signal, deviations from the intended straight-line survey path (z "bounce" and x-y "meander"), sensor pitch and roll error, errors in measurement of pitch and roll, GPS error, and sensor noise. All of the correlation structures for these noises were obtained from field data, including measurements of NRL's current ESTCP Project MM-0508.

3.4 Code Structure

The Monte Carlo code is written as a series of subroutines, each called in sequence during execution of the main loop. Each routine has access to a common data structure called "m" (for Monte Carlo), in which all records are stored. The "m" structure contains all inputs and outputs for each realization, including randomly determined values, all aspects of the target, the synthetic data set, the fit results, and final performance measures. The structure of "m" is a collection of arrays, each having length equal to the number of Monte Carlo realizations. The full definition of the "m" structure is included as Appendix A. We find this structure to be compact and flexible, and an easy way to access results for inspection and graphing after execution is finished.

3.4.1 *Code inputs*

For each individual realization of simulated data, the code accepts the inputs listed below (RMS means Root Mean Square):

TargSTR:	The class of UXO used to produce the synthetic data (string).
TargID:	The unique identifier of the specific UXO used (string).
DepthTarg:	The maximum burial depth for the target (m).
Betas:	The response beta values for the particular target, a (3 by 3) array.
Lane:	Survey lane spacing (m)
Speed:	Vehicle speed (m)
Samp:	Sensor sampling rate (1/s)
Hgt:	Nominal height from bottom of sensor to ground surface (m)
GPSxyErr:	RMS error of GPS measurements in x-y plane (m)
GPSxyErrCor:	Correlation length of GPS x-y errors (m)
GPSzErr:	RMS error of GPS measurements in z plane (m)
GPSzCor:	Correlation length of GPS Z errors (m)
GEOamp:	RMS geologic noise (sensor units e.g. mV for EM61mkII)
GEOcor:	Correlation length of geologic noise (m)
TIMEerr :	RMS timing error between EMI signal & navigation signal (s)

TIMEcor:	Correlation length of time error (s)
LANEamp:	Transverse RMS deflections off travel path (m)
LANEampCor:	Correlation length of transverse deflections (m)
BOUNCEamp:	RMS deflections in Z axis (m)
BOUNCEampCor:	Correlation length of Z deflections (m)
SENSerr:	RMS sensor noise. In sensor units (e.g. mV for EM61mkII)
SENScor:	Correlation length of sensor noise (s).
ROLLamp:	Sensor roll RMS amplitude (deg) for the true motion of the sensor.
ROLLampCor:	The correlation length of sensor roll (s) for the true motion of sensor.
ROLLerr:	Sensor roll RMS error (deg): the error in measuring roll.
ROLLerrCor:	The correlation length of roll errors (s).
PITCHamp:	Sensor pitch RMS amplitude (deg) for the true motion of the sensor.
PITCHampCor:	The correlation length of sensor pitch (s) for the true motion of sensor.
PITCHerr:	Sensor pitch RMS error (deg): the error in measuring pitch.
PITCHerrCor:	The correlation length of pitch errors (s).

3.5 Creating Synthetic Data

Synthetic data sets are produced using models of the processes responsible for real survey data. Using this approach we are able to draw realizations from the joint density function $pdf(\boldsymbol{X})$ without actually modeling $pdf(\boldsymbol{X})$ itself. Developing such a model would be infeasible in any case because of the large number of dimensions involved. The steps involved in creating synthetic data are as follows:

3.5.1 Create the Target.

UXO class is chosen (e.g., 20mm) and one individual UXO is selected randomly from that class to provide the beta response values. Target depth is chosen using a uniform distribution ranging from 0 (ground surface) to the maximum target depth for that class (Table 2). These depths are based on the 11 * target diameter rule of thumb [1]. Target x-y position is drawn from a uniform distribution, bounded within a box with side length equal to survey lane spacing, which ensures that all positions of the target relative to the survey lanes are possible. Target orientation is found by drawing from a uniform distribution on a sphere encompassing the target.

Ideally, we would have included clutter items in the library for the sake of completeness, however we do not feel it necessary, and do not expect their omission to affect results. Our focus is on quantifying how operational survey inputs (such as lane spacing and vehicle speed) affect accuracy of recovered betas, and not on the effect of specific target types. Performance is judged by comparing

UXO	Max burial depth
20mm projectile	0.25 m
60mm mortar round	0.8 m
81mm mortar round	0.9 m
3 inch Stokes mortar round	0.85 m

Table 2. Maximum target burial depths.

recovered betas against “true” values used to synthesize the data, across many random trials in a noisy environment. Goodness of fit values also are recorded for constrained inversion (fixed betas) to evaluate different weighting schemes on Chi-squared model errors.

There are also practical considerations in the decision to omit clutter. Any increase in library size would lengthen run times and reduce the total number of runs possible. We also had no readily available test stand data with the EM61 mkII on clutter. Rather than invent hypothetical values, or estimate values based on other sensors, we used betas that were readily available, which were only for UXO.

3.5.2 Create the Survey

Sensor measurement locations and orientations are generated. The procedure begins by locating way-points on a regular lattice at along-track intervals equal to half the data density, and at cross-track intervals equal to the lane spacing. This represents the intended path of the sensor: perfectly straight, evenly spaced lines. The way-points are then deflected in the transverse direction according to a random process with triangular correlation, with amplitude and correlation length specified by inputs. Finally, sensor measurement locations are found by following the path of the deflected way-points and interpolating sampling positions based on vehicle speed & sampling rate.

3.5.3 Create the Signals.

Given the “true” target parameters and sensor measurement locations, the forward model is run to produce the signals observed by the sensor. As mentioned in the executive summary, the dipole model was used to generate this synthetic data. The Dipole model has the advantages that it is fast and reasonably accurate for the majority of target/depth geometries, however the fact that it assumes the target to be a zero-volume point leads to some unrealism.

For real targets, recovered beta values depend on target depth and orientation, and this is entirely due to the fact that real targets have non-zero volume. Transmit and receive fields arriving at the nose of a real target are different from those arriving at the tail, both in strength and direction, and these fields change with target depth and orientation, producing changes in signal, and associated changes in apparent betas. At larger range, fields are more uniform throughout the body of the target, and this dependence is reduced. If the target is a zero-volume point, dependence disappears completely, i.e. there is no difference in nose-up and nose-down betas.

Since we use the dipole model for synthesis and analysis of EMI data, we could not simulate depth and orientation effects on recovered betas directly. Instead, we included two separate axial betas for each UXO item: a nose-up version, and a nose-down version. Essentially, these amount to additional library items. In follow-on work, we could incorporate non-dipolar effects in the synthetic data by using more sophisticated forward models, such as the BOR.exe numerical code from Dartmouth [7].

3.5.4 Add Survey Noise.

Geologic noise and sensor noise are generated and added directly to the synthetic data. Geologic noise comes from a 2-D random process correlated in x-y with triangular correlation. The observed sensor positions used for inversion are different from the true values used in the forward model. GPS errors in x-y and z are added (two separate correlated processes), and time errors are handled by deflecting each point a fraction of the way toward the following one, the fraction being the time error divided by sampling interval.

3.5.5 Invert.

Various kinds of inversion algorithms are included in the Monte Carlo tool. The options include either simultaneous or individual time-gate inversions, and a constrained, unconstrained, or hybrid beta approach. In all of these schemes, "Constrained" inversion refers to analysis in

Input	Description	Baseline value
1	lane spacing	0.5 m
2	speed of sensor	1. m/s
3	sampling rate	10 Hz
4	sensor height above ground	0.4 m
5	GPS xy error, RMS	0.4 cm
6	GPS z error, RMS	1.0 cm
7	Geologic noise, RMS	1.79 mV
8	timing error RMS	0.05 s
9	transverse deflections off travel path (meander), RMS	10.0 cm
10	vertical deflections off travel path (bounce), RMS	12.0 cm
11	sensor noise, RMS	0.2 mV
12	roll error, RMS	1 deg
13	pitch error, RMS	1 deg
14	roll amplitude, RMS	4 deg
15	pitch amplitude, RMS	4 deg
16	GPS xy error correlation length	250 s
17	GPS z error correlation length	250 s
18	Geologic noise correlation length	5 m
19	timing error correlation length	1.0e6 s
20	transverse deflections correlation length	6 m
21	deflections in z axis, correlation length (m)	6 m
22	sensor noise correlation length	0.1 s
23	roll error correlation length	2 s
24	pitch error correlation length	2 s
25	roll amplitude correlation length	2 s
26	pitch amplitude correlation length	2 s

Table 3. Baseline case inputs.

which a particular set of betas is assumed for the target. These betas are "archetypes", each representing a different class of UXO e.g. 20mm, 60mm mortar etc. The "archetypes" were formed for each class by taking the average at each time gate of all known betas in the class. So the archetypes are similar to each individual target beta, but identical to none. This is a realistic environment in which to test library-based discrimination schemes.

4. Results and Accomplishments

4.1 Baseline Case

The baseline case represents a survey using the Geonics EM61mkII sensor rolling on wheels over bumpy ground with an open view of the sky. Table 3 lists the baseline values for the various factors involved. Standard deviations and correlation scales for all twenty six inputs are taken directly from field experience, based on data described in the following pages. Synthetic versions of these inputs are guaranteed to possess the same desired standard deviation and correlation scale, thereby matching field experience.

Input 1: Lane Spacing: The spacing between intended straight-line survey lanes. The baseline value of 0.5 m reflects typical survey practice. Actual survey lanes follow a meandering path, which we simulate via random deflections off the intended straight line course. These deflections are produced by a correlated random process with RMS amplitude defined by input 9 and correlation length defined by input 20.

Input 2: Sensor Speed: Does not vary. The baseline value of 1 m/s reflects a comfortable walking speed.

Input 3: Sampling rate: Does not vary. Baseline value of 10 Hz is typical for the EM61mkII.

Input 4: Sensor height: Nominal sensor height above ground. Baseline value of 0.4 m reflects a typical wheel arrangement for the EM61mkII.

Input 5: GPS xy error: The RMS error of GPS measurements in x-y. The baseline value of 0.4 cm RMS may be too low. It is based on a long record of GPS data taken by Herb Nelson and Dan Steinhurst using a stationary GPS receiver (figure 3), but we are aiming here to simulate a moving receiver. Accurate assessment of error is difficult since field data from moving sensors invariably incorporate other sources (e.g., meander and bounce of the sensor). To get a good estimate of GPS errors for the moving case, a test in which the

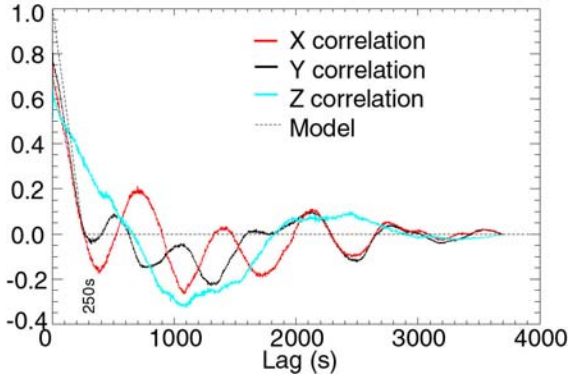

Figure 3. Correlation scales of GPS data using a stationary receiver.

receiver is translated precisely along a known path is needed. Such a test is planned under the NRL ESTCP Project MM-0508.

Input 6: GPS z error: Again, the baseline value of 1.0 cm RMS is based on the same stationary GPS dataset used for input 5, and may be too low for the moving case.

Input 7: Geologic noise: The baseline value of 1.79 mV is calculated from a dataset of EM61mkII measurements on the APG Blind Grid collected by Tetra-Tech Foster-Wheeler. We down-selected 35 patches of data from areas with no targets and no obvious spill-over signals from adjacent targets (Figure 4). Having no way to separate the different noise components in the data, we attributed the entire signal to geologic noise and assumed that all other sources were negligible. Although this assumption is obviously inaccurate, it provides a basic model for geologic variability, which, in any case, is expected to vary from site to site.

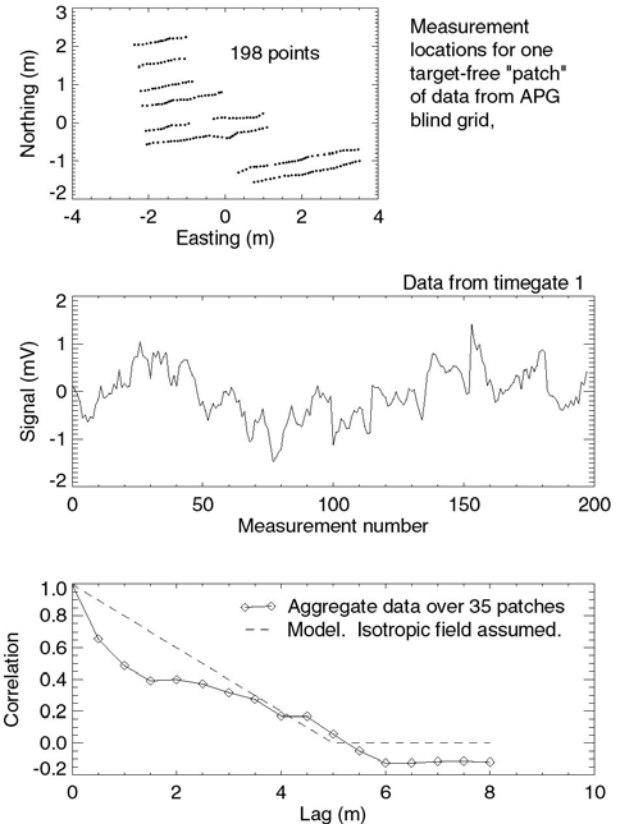

Figure 4. APG Blind Grid data is the basis for geo-noise input.

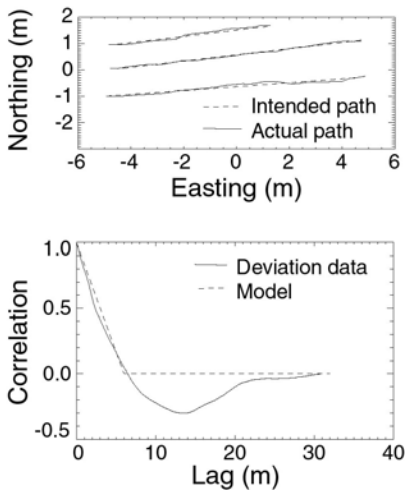

Figure 5. Deviations from the intended survey lane are based on data from Blossom Point.

Input 8: Timing error: This is the time lag between GPS and Sensor outputs. The baseline value of 0.05 s RMS is based on experience with various survey systems.

Input 9: Transverse deflection off travel path: The baseline value of 10 cm RMS is based on GPS data from surveys at Blossom Point. Sensor position data were broken up into individual survey lanes, and straight lines were then fitted to each using least-squares minimization. The straight line represents the intended survey path and transverse deviations (meander) were measured along the track. We assumed that the true path was close to the reported path (i.e., GPS errors were negligible) when deriving a value for this input (Figure 5).

Input 10: Vertical deflection off travel path: The baseline value of 12 cm RMS is based on the same GPS data from Blossom Point, and the same procedure as for input 9.

Input 11: Sensor noise: White noise with standard deviation 0.2 mV was independently added to each channel in the synthetic data. Without knowledge of the internal EM61 mkII electronics, or readily available data from a stationary sensor, we assumed the electronics produced similar noise levels in each channel. The value of 0.2 mV was based on EM61mkII data taken on the APG blind grid by Sky Research. In that dataset, standard deviation in each channel over target-free patches of ground was:

Channel 1: 1.79 mV
Channel 2: 1.08 mV
Channel 3: 0.56 mV
Channel 4: 0.33 mV

This represents a combination of noise sources: sensor noise, motion noise, and geologic noise. With no clear method of separating components, we made an educated guess that geologic noise dominated, and sensor noise was about $1/10^{th}$ as strong. This underscores the need for better characterization of these components, now underway in ESTCP project MM-0508 "Quantification of Noise Sources in EMI Surveys".

Inputs 12 & 13: Roll & Pitch error: This is the RMS error in the measurement of roll & pitch. The value of 1 deg is a rough estimate based on experience with Crossbow Inertial Measurement Unit (IMU) systems.

Inputs 14 & 15: Roll & Pitch amplitude: This is RMS error in measurement of roll & pitch. The value of 4 deg is an estimate based on a walking survey with a hand-held IMU sensor.

Inputs 16 & 17: GPS error correlation lengths: The value of 250 s comes from the stationary GPS data record used for inputs 5 & 6. See figure 3.

Input 18: Geologic noise correlation length: The value of 6 m comes from analysis of target-free patches of data from the APG blind grid (figure 4).

Input 19: Timing error correlation length: The value of 1.0^{6} s is meant to effectively hold timing error constant for each synthetic data set. Future work with this code may include lower values for timing error correlation when better data are available.

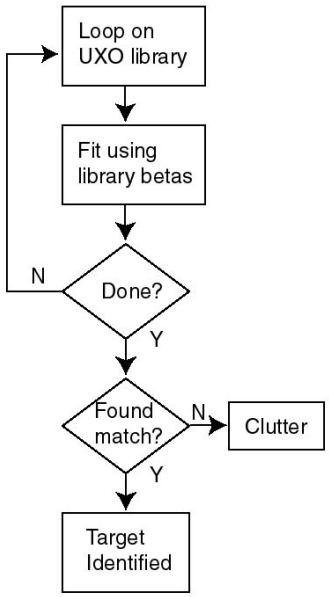

Figure 6. Constrained fitting for discrimination.

Input 20 & 21: Transverse and vertical deflection correlation length: The value of 6 m is based on sample correlation from a survey at Blossom Point (figure 5). Vertical correlation length was assumed to be similar to that in x-y.

Input 22: Sensor noise correlation length: The value of 0.1 s effectively makes this noise white, since the baseline sampling rate is 10 Hz. No good data on inherent noise in the EM61mkII were available to support a more precise correlation scale.

Inputs 23, 24, 25, and 26: Roll and Pitch error and amplitude correlation lengths: The value of 2 s is a rough estimate based on data from a hand-carried IMU. Correlation length for both true pitch variation and measurement errors were assumed to be the same.

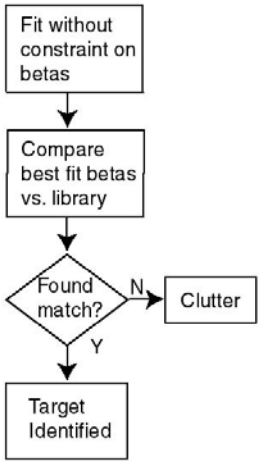

Figure 7. Unconstrained fitting for discrimination.

4.2 Realizations

Before any results were generated, the code was tested thoroughly to make sure it was working properly. Hundreds of realizations were synthesized and analyzed under zero noise conditions to make sure data was being inverted properly. Following the testing period, Two large sets of realizations were generated. The first set contained realizations of the baseline case without any variations, aimed at comparing different signal inversion approaches. The second run contained realizations of the baseline case using only the hybrid discrimination approach, but added variations in noise and survey configuration.

4.3 Evaluation of discrimination techniques

4.3.1 The Constrained approach

In this approach (Figure 6), we cycle through a library, doing a fresh fit for each item, with betas constrained to match library targets. Identification occurs when a good fit is found, and if none is found, then the target is not in the library. Usually the library contains only UXO, so targets not found in the library are identified as clutter.

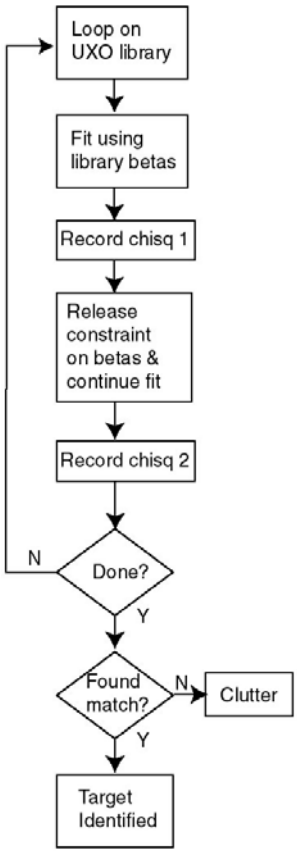

Figure 8. Hybrid approach to discrimination.

Experience has show that this approach has serious flaws. Defining what constitutes a "match" is problematic because chi-squared is strongly affected by factors that often vary from place

to place within a site. For example GPS navigation errors may be high near a tree line, causing chi-squared to be high for all targets there, even when fitting with the correct betas. Thus, it can be impossible to select a fixed chi-squared threshold for a match.

To overcome this problem, weighting schemes have been devised, in which chi-squared is scaled by some function that combines peak signal strength (SMAX) over the target and noise (RMS) estimated from a nearby patch of ground with no target. The weight defined as 1/(RMS^2 + (0.3 * SMAX)^2)) produced marked improvement on data collected with the GEM3 array at Blossom Point. [8]

4.3.2 The Unconstrained approach

In this approach (Figure 7) we invert the data without constraints on the betas. Inversion takes place only once, so run times are significantly faster. After inversion, the betas for the best fit are compared against the library and target identification occurs when a match is found. If no match is found, then the target is not in the library.

Experience has shown that this approach also has drawbacks. As with the constrained approach, defining what constitutes a match is difficult, since extraneous factors such as GPS error affect the chi-squared values and the best-fit betas, just as they do above.

In addition, this approach does not account for chi-squared surfaces with multiple local minima, each providing very different best-fit betas. A local minimum with UXO-like betas and a decent, but not globally minimal, chi-squared value is ignored by this approach, leading to false negatives.

4.3.3 The Hybrid approach

In this approach (Figure 8) we again cycle through the library doing fresh fits with constrained betas, but after each fit, we lift the constraints on the betas and perform a second fit. This second fit starts from the endpoint of the first fit. Chi-squared results from the second fit will always be equal to or better (i.e., lower) than results from the first fit, so the ratio (first chisq/second chisq) is always ≥ 1. Identification occurs when this ratio is close to 1. If identification does not occur, then the target is not in the library.

Ideally, the second fit should converge to the unique global minimum, no matter what the starting point; in practice it sometimes does, but it often finds different minima, and it's possible the global minimum might be missed altogether by the entire process. The reason we perform the unconstrained fit after each first fit is to guarantee that the best-ever chi-squared found through unconstrained fitting will be lower than all the values found through constrained fitting. This algorithm essentially produces a weight to be applied to chi-squared values from UXO library-constrained fits. The weight is simply 1/(global chisq minimum). It has the advantage that all extraneous sources of noise and error are incorporated such that each contributes to the weight in exact proportion to its influence on the chi-squared value. In this sense, it is optimal.

4.3.4 Procedure for discrimination

The Monte Carlo test bed was used to evaluate these fitting approaches in a run of 400 realizations using the baseline case survey parameters. All were simultaneous fits of the time gate response. Of these 400 realizations, 100 were made for each of the four UXO types in the study: 20mm, 60mm mortar, 81mm mortar, and 3in Stokes mortar. Each realization was submitted to inversion using all three methods above: constrained, unconstrained, and hybrid. The library contains 8 items: the four UXO types, each nose up and nose down.

In all cases, the library betas used are "archetype" betas, in the sense that each is supposed to represent the class of UXO. These "archetype" betas were produced by taking averages of measured betas for each UXO class. The specific betas used to create the synthetic data (i.e., the "true" betas) are randomly selected from the group of measured betas from real UXO, so they necessarily differ from the "archetypes".

Results are evaluated by comparing figures of merit for two groups: one group represents inversion using "right" betas (i.e., the library "archetype" betas are from the same UXO class as that used to generate the synthetic data), and the second group uses "wrong" betas (i.e., "archetype" betas from a different UXO class). The "wrong" group therefore corresponds to the real-world case where data from clutter items are analyzed using UXO library betas. In these cases, some evaluation of the mismatch is used to discriminate clutter. This approach is conservative for performance estimation, since we are using UXO betas to generate synthetic data and UXO betas to analyze the data, and are attempting to recognize differences due to one UXO versus another UXO. Since real clutter betas are often very distinct from UXO betas, we would expect real data from clutter to be more different (clutter versus UXO), making the job of discrimination easier.

Discrimination performance is measured by comparing results when the solver used the archetype for the same class of UXO as the true beta (i.e., used the "right" beta) versus cases where other archetypes were used. Since these methods cycle through all the archetypes in the library, there are many more runs (7) that use the wrong betas than runs (1) with the right betas.

4.3.5 Discrimination results

Figure 9 shows the results of the Monte Carlo runs in terms of the chi-squared values using the constrained approach. Results are shown for each UXO type as a function of whether the "right" or the "wrong" beta values were used. Better performance is defined by a lower chi-squared value and larger separation between the median values for the right and wrong betas. It is clear that using the right beta values produces a smaller chi-squared value, but the differential appears small.

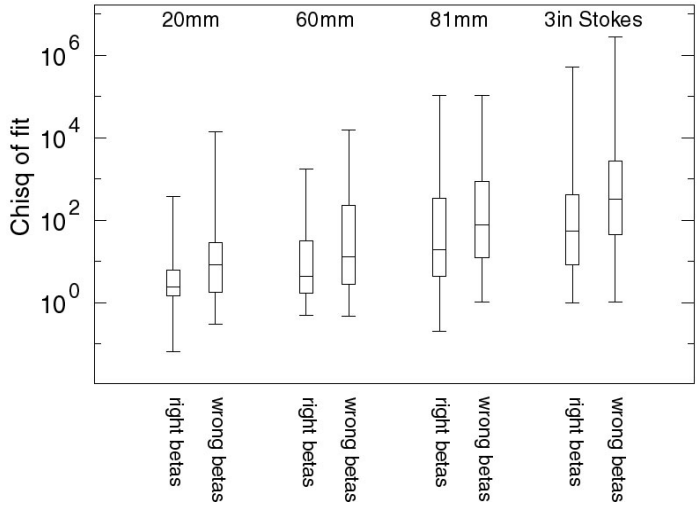

Figure 9. Constrained approach results for chi-squared values for each UXO type. Each box shows the upper and lower quartiles and the minimum and maximum, and the horizontal line denotes the median value. Each case represents 100 Monte Carlo realizations.

Figure 10 shows the results of the Monte Carlo runs in terms of the chi-squared values using the hybrid approach. Here, better performance is defined by a ratio of chi-squared values that approaches 1 and larger separation between the median values for the right and wrong betas. Compared to the constrained approach results shown in the previous figure, the hybrid approach produces larger separations between fits using the right and the wrong betas.

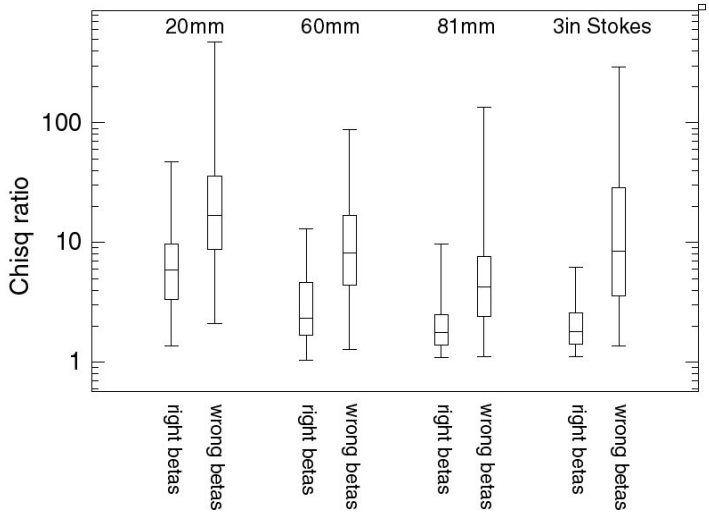

Figure 10. Hybrid approach results for chi-squared values for each UXO type. Each box shows the upper and lower quartiles and the minimum and maximum, and the horizontal line denotes the median value. Each case represents 200 Monte Carlo realizations.

The comparison between Figures 9 and 10 is one of different parameters and the improvement using the hybrid approach is not quantifiable. In Figure 11 below, we compare the same Monte Carlo run results for the constrained and the hybrid approaches using a better measure for comparison. In all three graphs, the red points were fitted with the beta values from the right class and the blue points were fit with beta values from the wrong class (as a simulation of data from clutter). Thus, for each value of maximum signal strength from a Monte Carlo run, there are 7 blue points and 1 red point (corresponding to the 8 items in the library). The measure of good performance for all approaches is a clear separation between red and blue points and a level or constant maximum chi-squared value across all signal strengths. The latter is important to be able use a fixed value of the measure for a discrimination threshold.

All three approaches show very poor separation at low signal strengths. The top graph for the constrained approach shows a mixing of results for almost all values of signal strength, and a strong upward trend of chi-squared values with signal strength. The middle graph, which shows the weighted chi-squared value (described in section 4.3.1), shows a bit better separation and less of an upward trend of chi-squared values with signal strength. The hybrid approach in the bottom graph, however, shows better separation for nearly all values of signal strength, a flat field of ratio values across signal strength, and even a trend of chi-squared ratio toward 1 for fits using the right beta class as the signal strength increases. In fact, many of the realizations produce a chi-squared ratio of 1 for relatively low signal strengths.

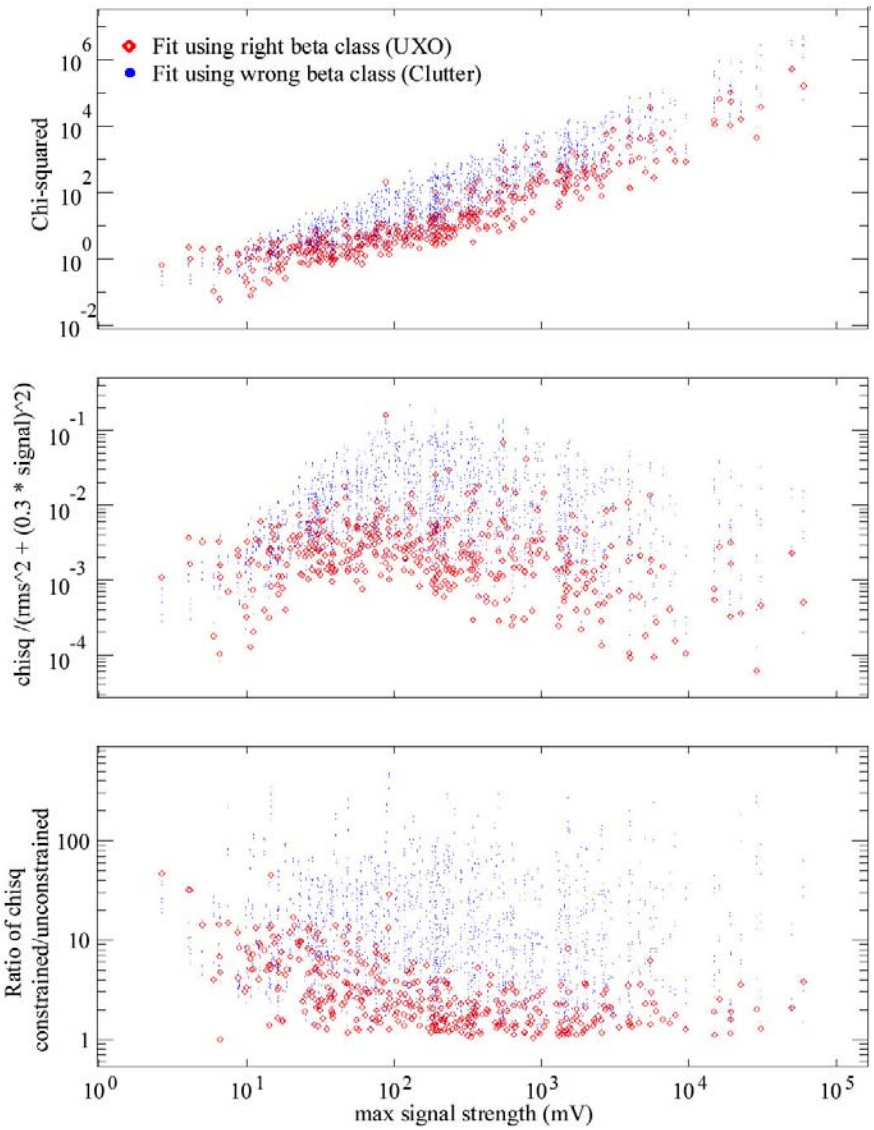

Figure 11. Measures of performance for the constrained approach (top), weighted constrained approach (middle) and hybrid approach (bottom), as a function of maximum signal strength of each realization. Red points were fitted with the beta values from the right class; blue points were fit with beta values from the wrong class.

The rising trend in the uppermost graph in figure 11 is related to the definition of Chi-squared error:

$$\chi^2 = \sum_{i=1}^{n} \frac{(d_i - s_i)^2}{n}$$

Where d_i and s_i represent data value and best-fit model value, respectively, at measurement location *i*. It is clear from this definition that higher signal strengths will result in higher chi-squared values.

4.4 Evaluation of noise errors and survey configuration sensitivities

The Monte Carlo test bed was used to evaluate the sensitivity of performance results to noise errors and survey configuration. The performance measures are the comparisons between fitted depth, horizontal location and dipole angle and truth (the values used for generation of the synthetic data). The discrimination technique used was straightforward unconstrained fitting with random re-starts, the best solution found assumed to be the global minimum.
The baseline case was the same one used for the investigation of discrimination techniques (see section 4.1). Excursions around the baseline case were made independently for lane spacing, survey speed, sample rate, sensor height, GPS xy error, GPS z error, geologic noise amplitude, timing error, lane deviations, bounce amplitude, sensor noise, roll amplitude, pitch amplitude, roll error (i.e., error in measuring roll amplitude), and pitch error. Each excursion from the baseline case was represented by 200 Monte Carlo realizations.

Results are shown in Figures 12 through 15. In each figure, the boxes show the upper and lower quartiles and the horizontal line in each box denotes the median (the maximum and minimum values could not be shown on these graph scales). The dashed line is the median of the baseline case.

The results show some interesting trends. In general, excursions from the baseline that improve system specifications (e.g., smaller noise errors, smaller lane deviations) decrease the errors and increase the coherence of the fits and vice versa for looser system specifications. Some factors show strong sensitivity to changes; the most sensitive appears to be timing error. Timing errors produce a characteristic "chevron" or "Herring-bone" pattern in the data due to alternating directions on successive survey lanes. When this pattern is recognized, timing error can be reduced or even removed during data processing through simple trial-and-error adjustment.

Occasionally, results from the Monte Carlo runs are opposite to those expected; examples are the effect of larger GPS z error and bounce/pitch amplitude error on x-y target position error, and several others on target depth error. These may be due to the number of Monte Carlo runs used (200 per case) and will have to be checked in the future. Some of the “unexpected” results are likely real, such as the decrease in coherence of fits for decreased lane spacing. This result was also observed in ESTCP project 200210 "Feature-Based UXO Detection & Discrimination", for which several causes have been postulated. For one, lower amounts of data over an anomaly allow the model to achieve good coherence using inaccurate target parameters. The many unexpected trends with excursion direction for the fitted angle errors is an indication that these angles are not being fitted well under any of the survey conditions used.

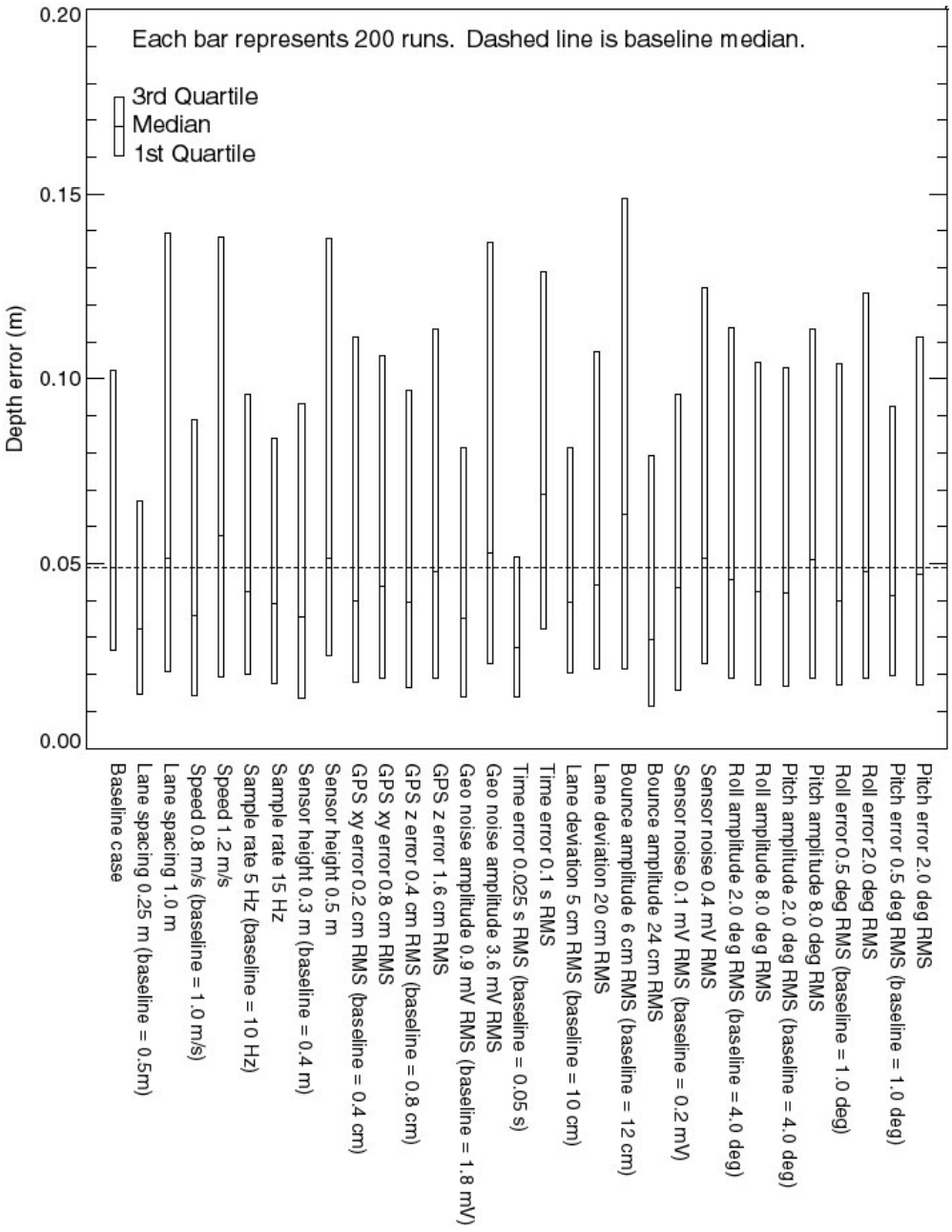

Figure 12. The sensitivity of fitted depth errors to changes in survey parameters.

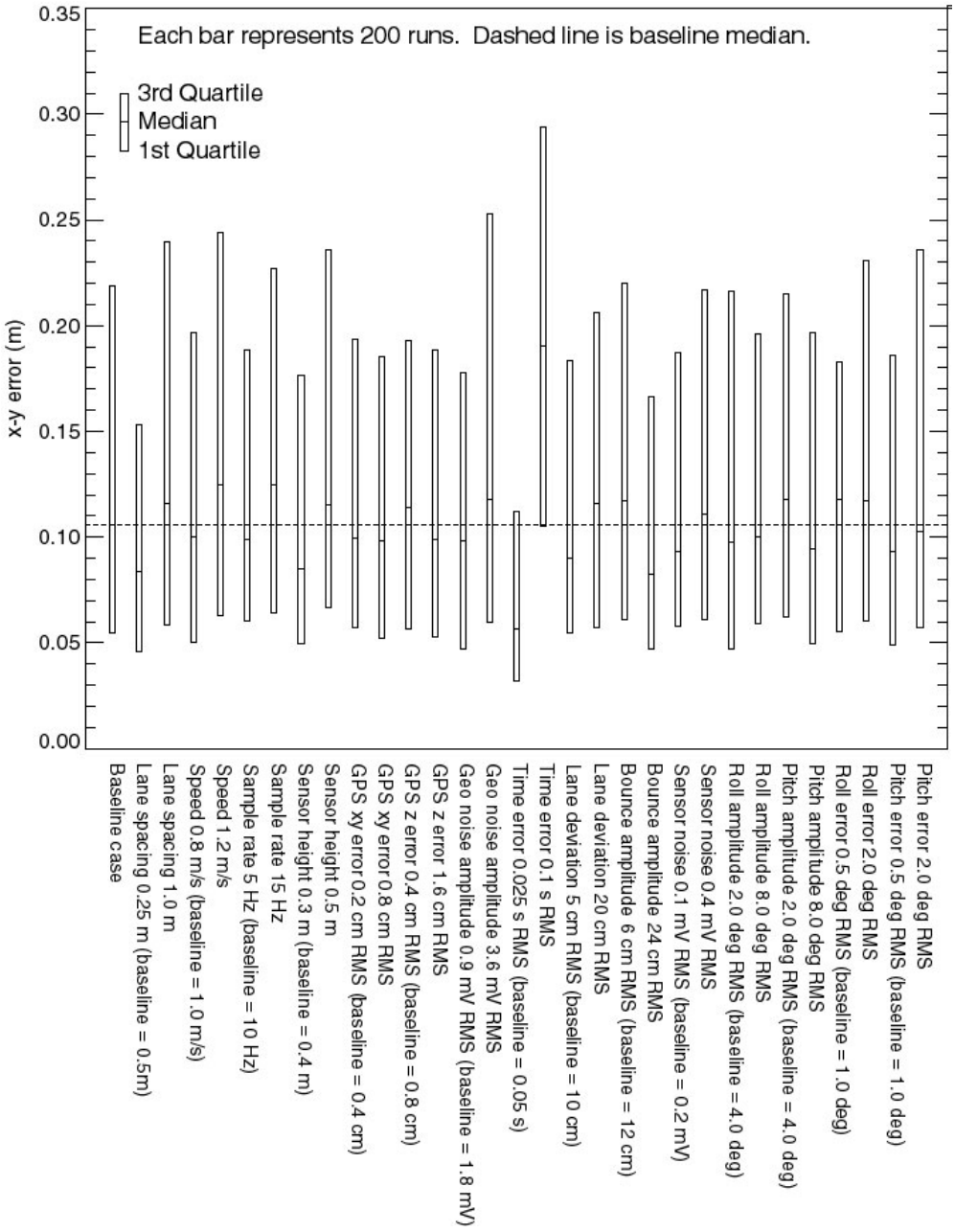

Figure 13. The sensitivity of fitted x-y position errors to changes in survey parameters.

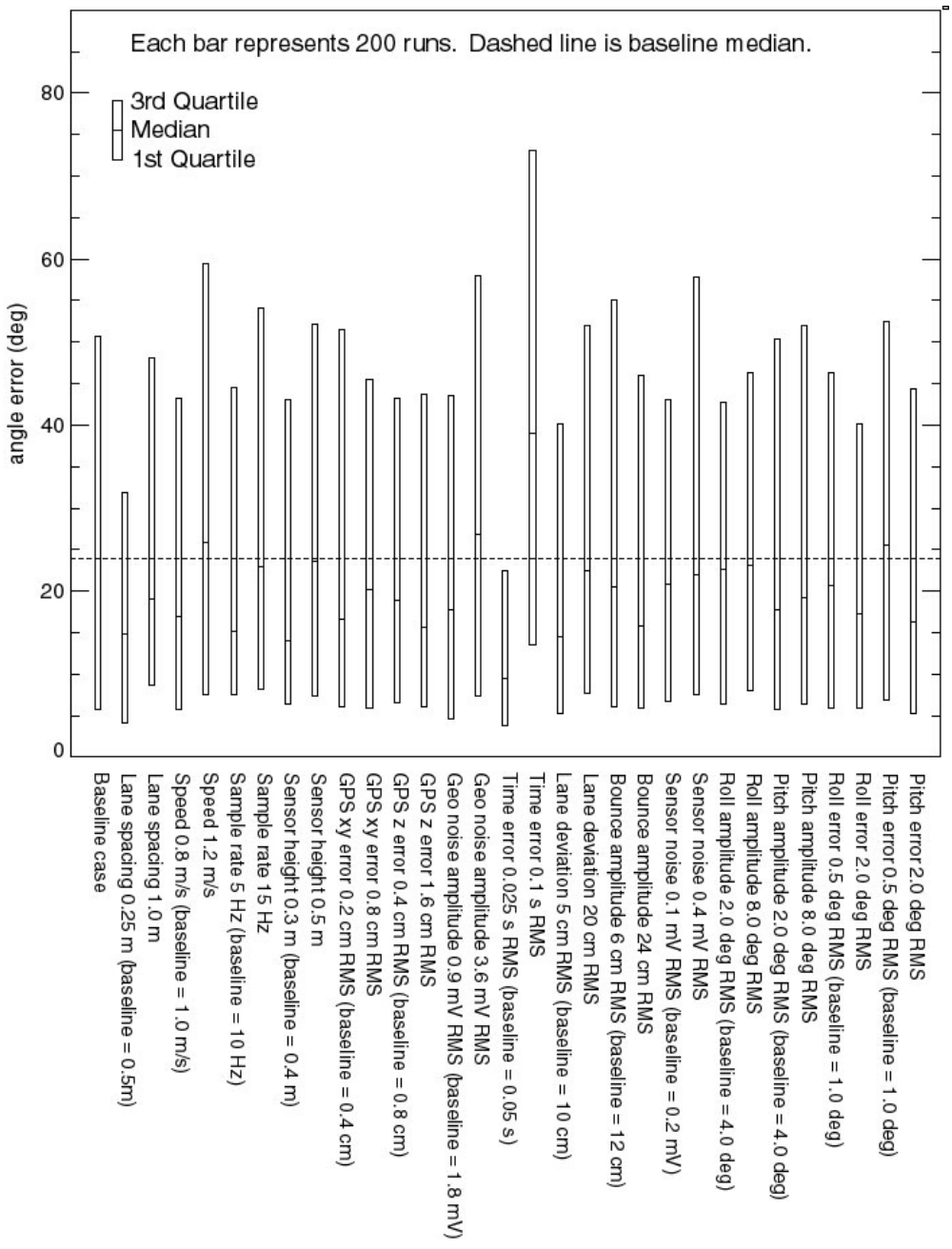

Figure 14. The sensitivity of fitted angle errors to changes in survey parameters.

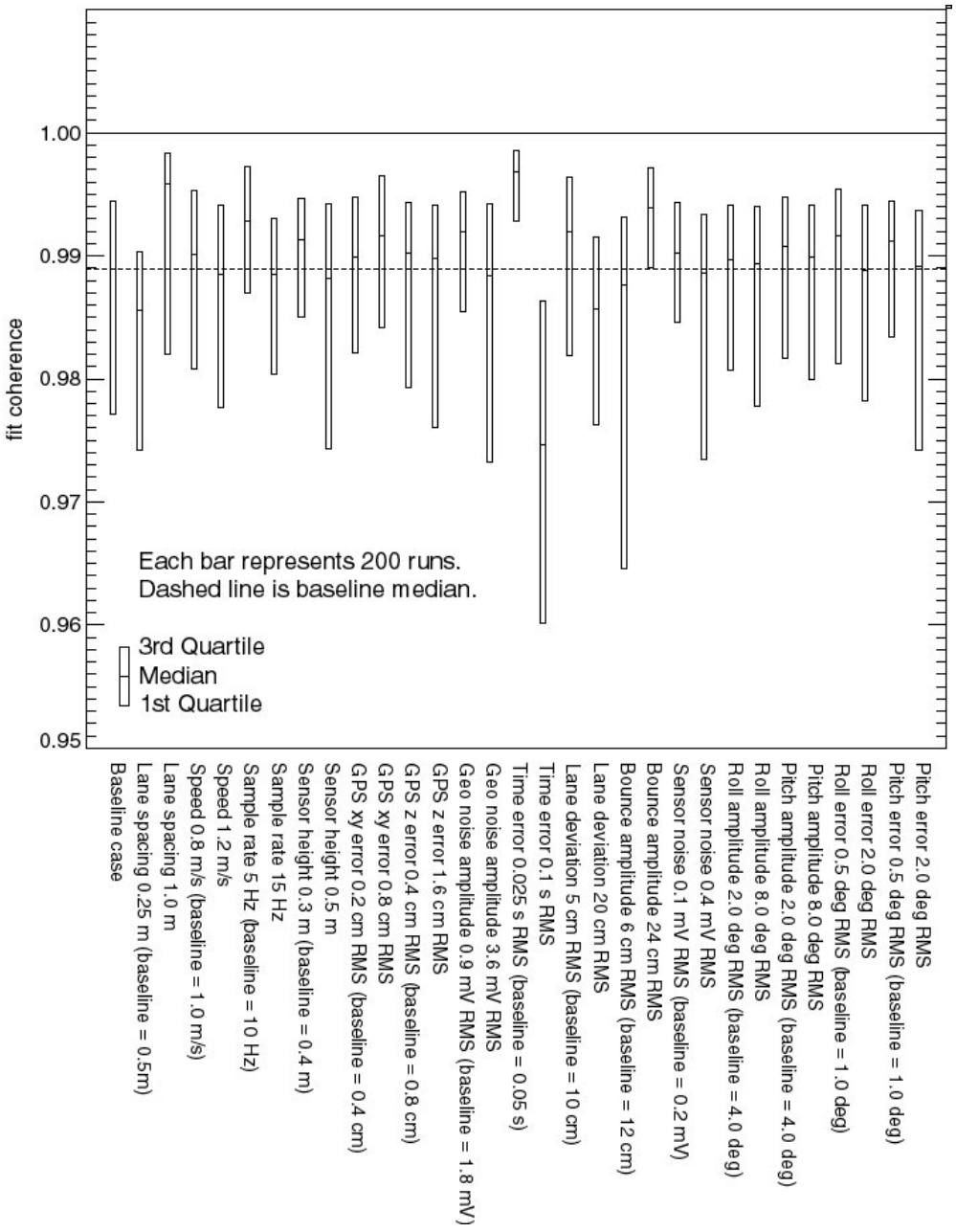

Figure 15. The sensitivity of coherence of fitted model to changes in survey parameters.

5. Conclusions

This project has produced a Monte Carlo tool that calculates performance measures under any given set of survey conditions and analysis methods. Improved noise models and discrimination technique options were incorporated, and proof-of-principle has been shown that this tool can provide relevant information and guidance on critical areas to improve in field surveys, and suggestions for improved discrimination algorithms. Even under the limited set of Monte Carlo realizations performed for this proof-of-principle, indications are clear that the hybrid discrimination approach is well worth further investigation, and improving system timing errors can provide large improvements in performance.

6. References

[1] Bob Selfridge, Personal Communication at the SERDP/ESTCP EMI Workshop, February 4, 2004, Annapolis, MD

[2] W.H. Press, S. Teukolsky, W. Vetterling, and B. Flannery, Numerical Recipes in C, Cambridge University Press, New York, NY, 1997.

[3] AETC, Incorporated, "Processing Techniques for Discrimination Between Buried UXO and Clutter using Multisensor Array Data," SERDP Project UX-1121 Annual Report, December 2000.

[4] Erik VanMarke, Random Fields MIT Press, Cambridge Mass, 1988.

[5] Edward Isaaks and R. Mohan Srivastava (1989) Applied Geostatistics. Oxford University Press, New York, NY.

[6] Jonathan Miller (1994) *Composite Random Field Models in Environmental Risk Assessment*. Master of Science in Engineering degree thesis. Princeton University, Princeton NJ.

[7] Sun, Keli, Kevin O'Neill, Fridon Shubitidze, Irma Shamatava, and Keith D. Paulsen (2005) *Fast data-derived fundamental spheroidal excitation models with application to UXO discrimination*. IEEE Transactions on Geoscience and Remote Sensing, 43(11): 2573-2583.

[8] Nagi Khadr, Personal Communications 2006.

Appendix A: The Common Data Structure

```
pro initializeStructures
;This routine creates the data structure used in the Monte Carlo loop.  It does not populate it  with information.

;'m' is a structure of arrays, each array is the same length.  Some of these are pointer arrays.  Each line in these
; arrays is a record of a single monte carlo run.  For example, m.xtru(37) and *m.s(37) record the true x coordinate and
; the observed signal for the 37th run.

; phi,theta,psi:       yaw, pitch, and roll of target (degrees, scalars).
;                  To understand exactly what these angles mean, consider the target as an airplane pointed
;                  initially in the +X direction (East).  Wings are level,
;                  aligned with Y axis (North-South), with left wing in +Y direction.
;                  Pilot's spine aligned with Z axis (up-down) with pilot's hat in +Z direction.
;
;                  Now follow these steps:
;                  * First, phi (yaw) rotates plane about Z axis.  Positive angle => CCW.
;                  * Next, theta (pitch) rotates plane about Y' axis - that's the axis thru
;                      the current position of the wings - not the initial Y axis.
;                      Positive angle => nose down.
;                  * Third, psi (roll) rotates the plane about the X'' axis - that's the axis
;                      thru the current position of the fuselage - not the initial X axis.
;                      Positive angle => right wing 'drops'. (the sense of the word 'drops'
;                                              applies when theta is small)
;
;                  These output angles are always bound by these limits:
;                      phi   => (0 to 360)
;                      theta => (0 to 90)  means target is always nose-down.  Not a problem since
;                                  dipole model assumes no "nose" or "tail" - both the same.
;                                  "Nose up" targets are represented by adding 180deg to phi.
;                      psi   => (-90 to 90)

; b1, b2, b3:          Axial response betas.  Vectors dimension (nGate).  Using the airplane analogy above,
;                  b1 informs response along the fuselage,  b2 along the wingspan, and b3 along the pilot
;                  spine.
;

common mCommon, m       ;m is the data structure that records all the monte carlo results
common datCommon, nChan, nGate, nMonte, loopTarg, depthTarg, loopN, sensor, ik, radiusTarg, chanAtten
                  ;nChan is the number of channels (time or frequency).
                  ;nGate is the number of gates (time or frequency).
                  ;nMonte is the number of runs in the study, (long int)
```

```
                ;loopTarg is the list of targets (string array)
                ;depthTarg is an array of max burial depths (dbl)
                ;loopN is the number of realizations per target (integer)
                ;sensor is the sensor type (string)
                ;ik is the uber-alles counter that keeps track of current iteration (long int)
                ;radiusTarg is the default radius of anomaly for data carving
                ;chanAtten is an array giving the relative strengths of geonoise in each channel.
common baseline, baseLane,baseSpeed,baseSamp,baseHgt, $
    baseGPSXYerr,baseGPSZerr,baseGEOamp,baseTIMEerr,baseLANEamp,$
    baseBOUNCEamp,baseSENSerr,baseROLLerr,basePITCHerr, $
    baseROLLamp, basePITCHamp, $
    baseGPSXYcor,baseGPSZcor,baseGEOcor,baseTIMEcor,baseLANEcor,$
    baseBOUNCEcor,baseSENScor,baseROLLerrCor,basePITCHerrCor, $
    baseROLLampCor, basePITCHampCor                ;The baseline configuration stored here.

; Set up structures to be used by tu & tc below
; s is simultaneous fit to all timegates
s = {x:0.0d,y:0.0d,z:0.0d,phi:dblarr(3),theta:dblarr(3),psi:dblarr(3), $
   betas:dblarr(3,3),coh:0.0d,chisq:0.0d}
; i is each timegate fit individually
i = {x:dblarr(3),y:dblarr(3),z:dblarr(3),phi:dblarr(3), $
   theta:dblarr(3),psi:dblarr(3),betas:dblarr(3,3),coh:dblarr(3),chisq:dblarr(3) }
t = {s:s,i:i,id:''} ; structure for one call

m = { $

          ;SURVEY PARAMETER INFORMATION - a record of inputs to the runMonte() routine

        Lane       :dblarr(nmonte) $; spacing between survey lanes (m)
      , Speed      :dblarr(nmonte) $; speed of sensor travel (m/s)
      , Samp       :dblarr(nmonte) $; sampling rate of sensor (samples/s)
      , Hgt        :dblarr(nmonte) $; nominal height from bottom of sensor to ground surface (m)

          ;ERROR INFORMATION - a record of the inputs to the runMonte() routine

      , GPSXYerr   :dblarr(nmonte) $; RMS error of gps positioning (x & y) (m)
      , GPSZerr    :dblarr(nmonte) $; RMS error of gps positioning (z) (m)
      , GEOamp     :dblarr(nmonte) $; RMS amplitude of true geologic noise (sensor units - e.g. mV for EM61mkII)
      , TIMEerr    :dblarr(nmonte) $; RMS error of timing offset
      , LANEamp    :dblarr(nmonte) $; RMS amplitude of deflections about travel path (m)
      , BOUNCEamp  :dblarr(nmonte) $; RMS amplitude of true deflections in z axis.
      , SENSerr    :dblarr(nmonte) $; RMS error of sensor noise (sensor units e.g. mV).  Const across all gates.
      , ROLLamp    :dblarr(nmonte) $; RMS amplitude of true roll (deg).
```

```
    , ROLLerr     :dblarr(nmonte) $; RMS error of roll (deg).
    , PITCHamp    :dblarr(nmonte) $; RMS amplitude of true pitch (deg).
    , PITCHerr    :dblarr(nmonte) $; RMS error of pitch (deg).

       ;TARGET INFORMATION

    , xTru        :dblarr(nmonte) $; true x,y,z coordinates of target (m)
    , yTru        :dblarr(nmonte) $;
    , zTru        :dblarr(nmonte) $;
    , phiTru      :dblarr(nmonte) $; true phi, theta, psi euler angles of target
    , thetaTru    :dblarr(nmonte) $;
    , psiTru      :dblarr(nmonte) $;
    , betaTru     :ptrarr(nmonte) $; true target betas, pointers to dblarr(nGate,3) arrays.
    , targStr     :strarr(nmonte) $; string describing the true target e.g. '81mm'
    , targetID    :strarr(nmonte) $; integer ID for target, corresponds to objID in d structure.

       ;SURVEY INFORMATION

    , nGrid       :intarr(nmonte) $; the number of survey data points in the grid
    , xGridTru    :ptrarr(nmonte) $; true x,y,z coordinates of survey grid locations - pointer to dblarr(nGrid) arrays (m)
    , yGridTru    :ptrarr(nmonte) $;
    , zGridTru    :ptrarr(nmonte) $;
    , rollTru     :ptrarr(nmonte) $; true roll, pitch & yaw of sensor - pointer to dblarr(nGrid) arrays (degrees)
    , pitchTru    :ptrarr(nmonte) $;
    , yawTru      :ptrarr(nmonte) $;
    , sTru        :ptrarr(nmonte) $; true signal produced at the true grid points - pointer to dblarr(nGrid,nGate) (sensor units e.g. mV for
          EM61mkII)
    , xGridObs    :ptrarr(nmonte) $; Observed x,y,z coordinates of survey grid locations - pointers to dblarr(nGrid) arrays (m)
    , yGridObs    :ptrarr(nmonte) $;
    , zGridObs    :ptrarr(nmonte) $;
    , rollObs     :ptrarr(nmonte) $; Observed roll, pitch & yaw of sensor - pointers to dblarr(nGrid) arrays (degrees)
    , pitchObs    :ptrarr(nmonte) $;
    , yawObs      :ptrarr(nmonte) $;
    , sObs        :ptrarr(nmonte) $; Observed signal, after sensor noise added - pointer to dblarr(nGrid,nGate) (sensor units e.g. mV for
          EM61mkII)

       ;INVERSION INFORMATION

    , tc          :replicate(t,nmonte,2*n_elements(loopTarg)) $; array of structures that stores the constrained fit results
    , tu          :replicate(t,nmonte,2*n_elements(loopTarg)) $; array of structures that stores the unconstrained fit results.

    }
end
```